802.11b and associated network security risks for the home user

by Michael Osten
mosten@bleepyou.com

Background

Approved in 1997 by the IEEE 802 committee, 802.11 details the framework necessary for a standard method of wireless networked communications. 802.11 uses the 2.4-GHz microwave band designated for low-power unlicensed use by the FCC in the USA in 1985.

It allows for two distinct methods of encoding, FHSS and DSSS. FHSS (Frequency Hopping Spread Spectrum) distributes the communication across 75 one-MHz subchannels, randomly skipping between them.

Two operating modes are defined: infrastructure and ad hoc. Dedicated hardware (the access point) provides a basic or extended service set that builds the wireless infrastructure. It provides basic bridging, as well as allowing clients to roam from access point to access point (provided they all exist on the same ethernet segment; roaming across subnets is not possible at this time). The ad hoc (IBSS, or Independent Basic Service Set) mode allows individual nodes to participate in a peer-to-peer network without an access point.

DSSS (Direct Sequence Spread Spectrum) breaks the band into 14 overlapping 22-MHz channels and uses one channel at a time.

In September of 1999, the 802 committee extended the specification, deciding to standardize on DSSS. This extension, 802.11b, allowed for new, more exotic encoding techniques. This pushed up the throughput to a much more respectable 5.5 Mbps (up to 11 Mbps). While breaking compatibility with FHSS schemes, the new extensions made it possible for new equipment to continue to interoperate with older 802.11 DSSS hardware.

The Wired Equivalent Privacy (WEP) algorithm is used to protect wireless communication from interception. A secondary function of WEP is to prevent unauthorized access to a wireless network; this function is not an explicit goal in the 802.11 standard, but it is frequently considered to be a feature of WEP. WEP relies on a secret key that is shared between a mobile station (e.g. a laptop with a wireless ethernet card) and an access point (ie. a base station). The secret key is used to encrypt packets before they are transmitted, and an integrity check is used to ensure that packets are not modified in transit. The standard does not discuss how the shared key is established. In practice, most installations use a single key that is shared between all mobile stations and access points. More sophisticated key management techniques can be used to help defend from the attacks we describe; however, no commercial system I am aware of has mechanisms to support such techniques.

Problem

802.11b is an increasingly common method of distributing network connectivity to mobile users in both the workplace and the home. Unlike corporate users, little attention has been paid by home users to the security of personal wireless networks.

Most consumer hardware access points have WEP capabilities, but as my research has shown very few home wireless "administrators\ enable WEP, disable broadcast of SSID, or even change the default settings that shipped from the factory.

Even with WEP enabled, users can not be certain of network security as outlined in "Security of the WEP algorithm\u" paper detailing the faults of WEP's RC4 encryption algorithm
It has become fairly trivial to break WEP with several hours of wireless sniffing with software designed for this purpose. (See Resources)

This oversight leaves users open to unauthorized access to internal privileged information, as well as an unauthorized use of network services.

Research

802.11 Access point detection:

Over the course of several days I have mapped hundreds of residential and business wireless access points in the Ft. Lee/Jersey City/Hoboken New Jersey areas. (see figure 1-3). All captures were from a car traveling at hiway speeds. Less than 2 percent have enabled WEP. Less than 20 percent have changed the default SSID from factory settings. This causes a problem in that a person with a small amount of hardware can hijack broadband connection with total anonymity from a car, or an apartment across the street. All data presented here was collected with a Compaq Ipaq and Windows CE with a Lucent WavLan 802.11b WIFI card running Ministumber (See Resources).

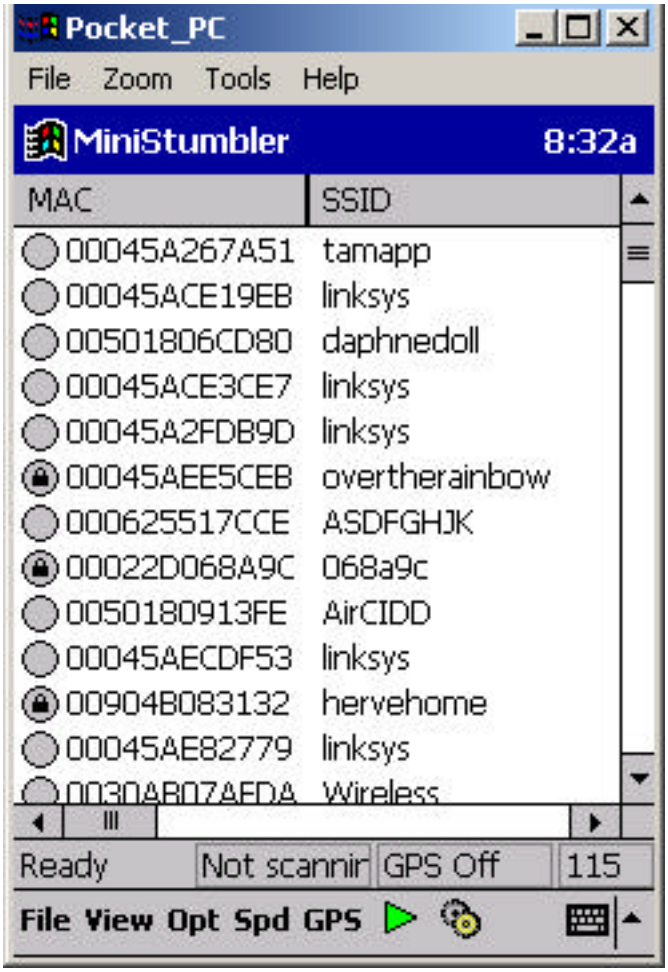

Figure 1 (Ministumber displaying detected SSID's)

WEP and weakness of RC4 key scheduling:

All data presented here was collected with a Sony Viao using a D-Link Prism2 based 802.11b card, and Linux using “Kismet\ “Ethereal\and \AirSnort\See Resources) to demonstrate breaking WEP. All data collected in the attempt to demonstrate the weakness of WEP originated from my personal network.

There are several dangers to having not enabled WEP. Although WEP
has proven insecure, it still provides a deterrent for all but the most determined intruder.

Figure 3 and 4 show what happens when 802.11b traffic is sniffed using Kismet and Ethereal. All packets are logged, and all plain text login/passwords (telnet, pop3, ect) are clearly shown by looking through the logs as displayed by Ethereal. As well as traffic history, what websites you have visited, etc.

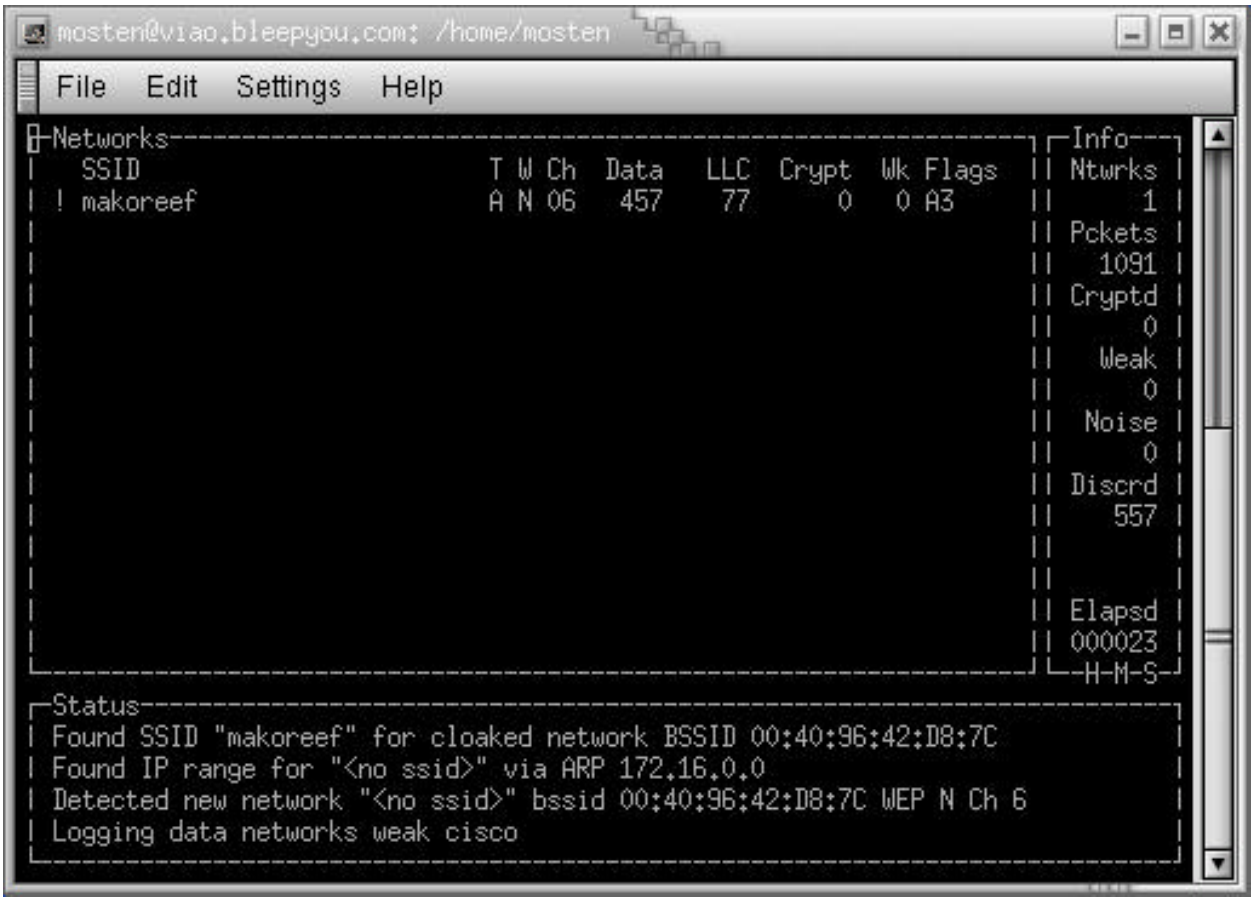

Figure 3 (Kismet monitoring non WEP enabled network)

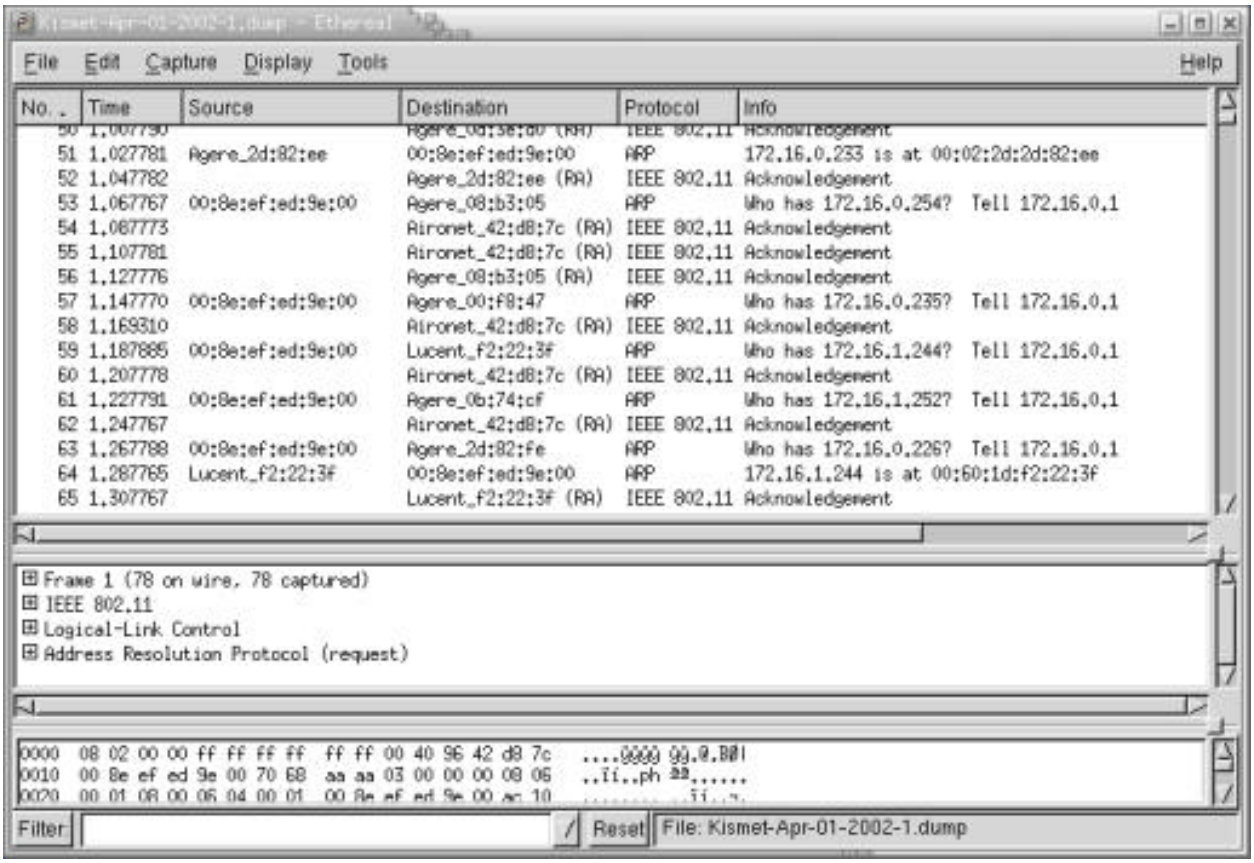

Figure 6 (Ethereal with logs imported from Kismet)

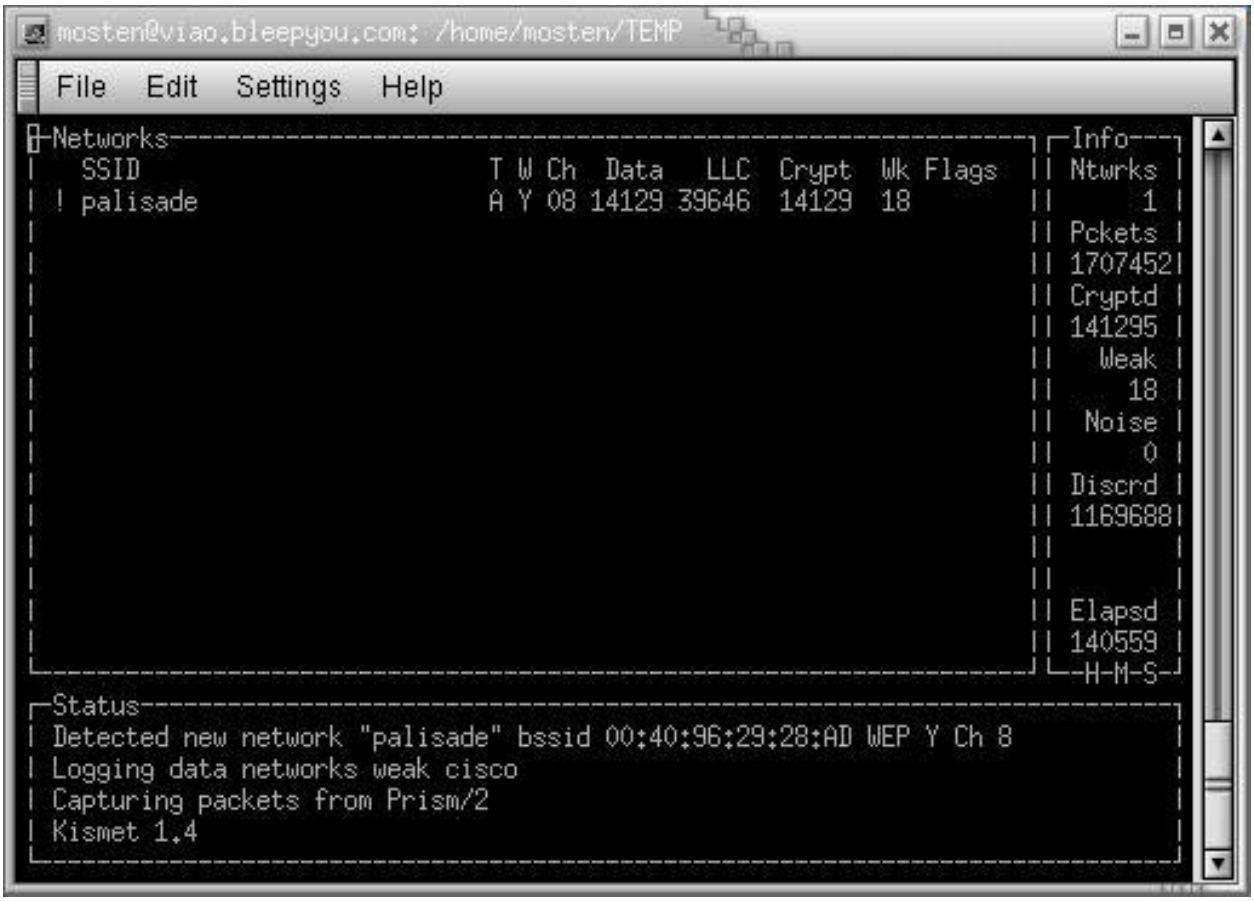

Figure 4 (Kismet monitoring WEP enabled network)

Notice the "Crypt" field of figure 4. This field displays the actual number of WEP encrypted packets that have traveled from the 802.11b access point. 802.11b access points can be very noisy, sending broadcasts to wireless clients several times a second. These broadcasts are not WEP encrypted, numbers of which can be seen under the "Data" heading. Weak RC4 keys under the "Wk" heading are the significant field when attempting to break WEP security. Airsnort requires 1500-1800 weak (Wk) RC4 encrypted packets to extract the correct 128 WEP key. This would require approx. 4 million WEP encrypted packets.

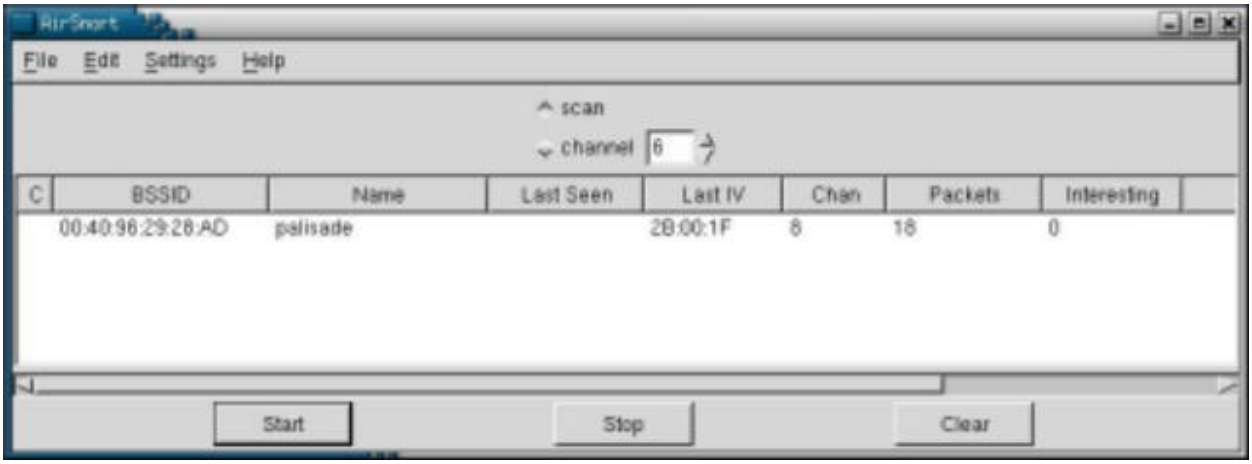

Figure 4 (AirSnort monitoring WEP encrypted network)

Conclusions and proposed possible solutions

First and foremost, WEP, although insecure, provides a significant deterrent to the casual easedropper. Consider that exploiting the weaknesses of WEP would require 8+ hours of sniffing and millions of packets of encrypted data.

Consider installing a firewall that separates your wireless access point from any internal network. This would reduce the risk of sensitive data being intercepted via sniffing, or exploits run on internal machines.

NoCatAuth (see resources) provides both user authe ntication and firewall capabilities for wireless networks, and is used extensively in the open wireless community. NoCatAuth is licensed under the GPL making source code available to the user.

Some of the key features of NoCatAuth:

- User Authentication
- Statefull firewalling
- Caching DNS server
- Quality of Service

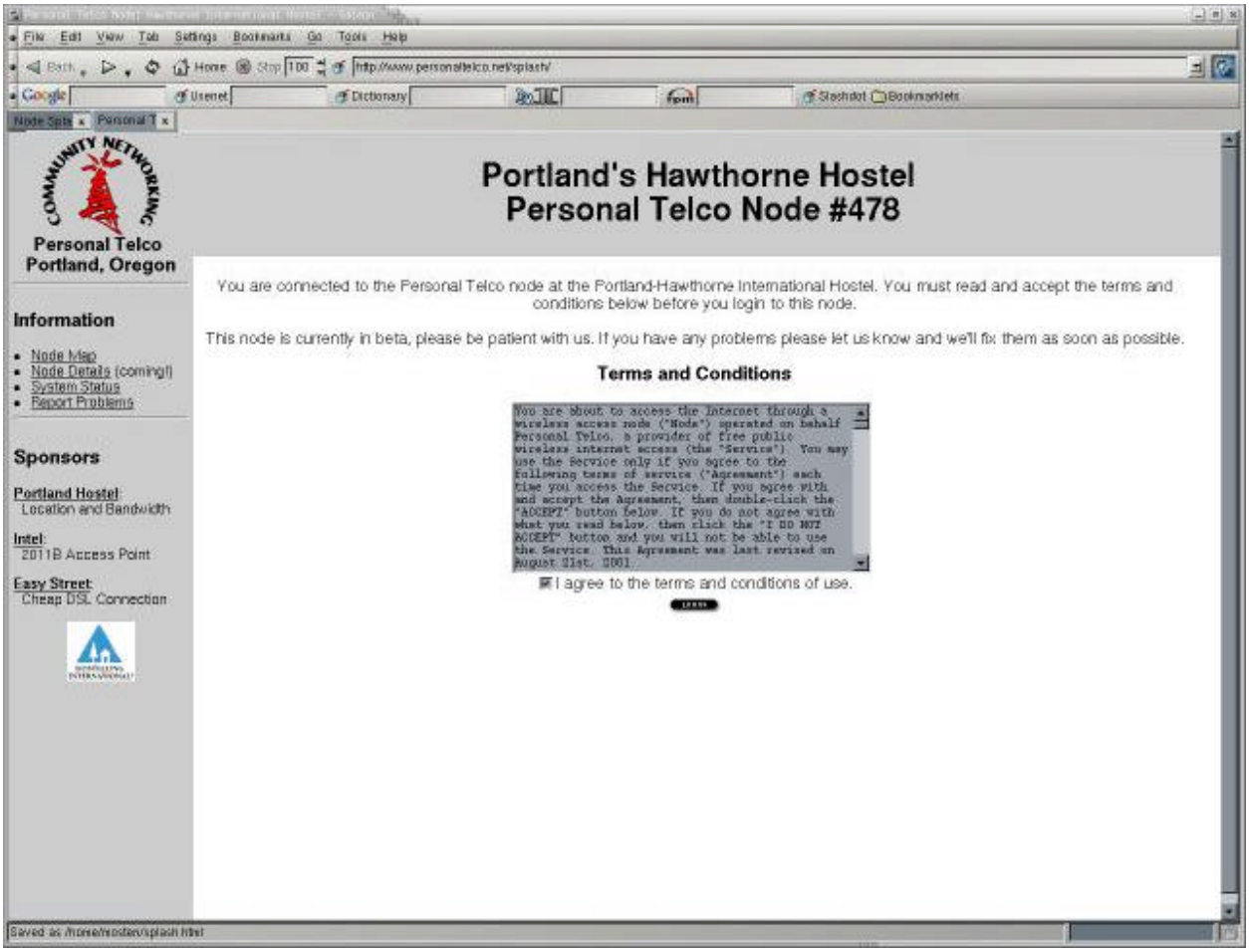

Figure 5 (NoCatAuth login page from Portland Hawthorne Hostel)

Resources

=======

Security of the WEP algorithm\u" (http://www.isaac.cs.berkeley.edu/isaac/wep-faq.html)
"NetStumbler\d \u"Ministumbler\u" (http://www.netstumbler.com/)
Kismet (http://www.kismetwireless.net/)
NoCatNet (http://nocat.net/)
Ethereal (http://www.ethereal.com/)